

This project was the part of the course “14EC601 - Machine Learning ” at PES Institute of Technology, Aug - 2017

**Problem Statement:** Developed a simple sentiment classifier based on a public set of texts labeled with various sentiments, like happiness, love, surprise, etc using Bayesian classifier

1. Training of the simple classifier:

- a. A dataset from CrowdFlower containing tweets and sentiment labels was used.
- b. We used a model from Naive Bayes as it's quite simple and efficient compared to SGD, SVM and RandomForestClassifier.
- c. Sklearn was used for encoding the data.
- d. For feature extraction, sklearn's TfidfVectorizer helps in assigning weights to words according to their inverse document frequency
- e. Words that occur often get a lower weight since they tend to be less informative.
- f. LabelEncoder assigns unique integers to the different labels it sees
- g. Construct the Bayesian model and evaluate it.

2. Testing the simple classifier:

- a. Bayesian models assume that the contribution of each word is independent of the other words, we can input the individual words to check its analysis.
- b. As the model expects a series of texts, each encoded as a vector whose length is equal to the size of the vocabulary.

**Result:** Achieved precision score for 28% using MultinomialNB for vocabulary size of 40000, on contrast we also trained a CNN and achieved accuray 38% for the same data set.

**Simple Flask App for sentiment analysis:**

1. Used a pipelined Logistic Regression Model to classify the tweets and deployed that model using Flask
2. Wrote a Python script to scrape the tweets related to a particular text query
3. A user interface where the user can submit his query
4. Once we get the query from the user, we will use the twitter API to get the tweets related to the searched query
5. The saved model to predict the class of the tweets and send the results back to the webpage

